Gating Artificial Neural Network based Soft Sensor

Petr Kadlec and Bogdan Gabrys

Computational Intelligence Research Group
Bournemouth University
Fern Barrow, Poole
BH12 5BB
United Kingdom

Abstract. This work proposes a novel approach to Soft Sensor modelling, where the Soft Sensor is built by a set of experts which are artificial neural networks with randomly generated topology. For each of the experts a meta neural network is trained, the gating Artificial Neural Network. The role of the gating network is to learn the performance of the experts in dependency on the input data samples. The final prediction of the Soft Sensor is a weighted sum of the individual experts predictions. The proposed meta-learning method is evaluated on two different process industry data sets.

1 Introduction

Modern production plants in the process industries are extensively instrumented with the data primarily recorded for process control purposes. But in recent years the data has found another form of application. Drawing upon techniques from statistics, pattern recognition, and machine learning, the data is being used to build predictive models which are within the process industry called *Soft Sensors*.

There are several reasons for the interest of the process industry in the development of data-driven Soft Sensors. One of the most important reasons is the difficulty in the development of model-driven Soft Sensors like First Principle Models (FPM). FPMs usually take a form of mathematical equations which make use of the knowledge of the physical and chemical laws for building models of the processes. Remarkably both static (energy or mass balance based) and dynamic simulators exist but the processes are usually too complex to be correctly and precisely described. There are also lots of external influences, e.g. the environmental temperature, the purity of the educts, the abrasion of different mechanical parts, which make the modelling of the exact process dynamics very difficult. For these reasons, the models have to be abstracted from the reality and focus on the important aspects of the process. An alternative way to make predictions about the state of the process or the product quality is to use the data, which is measured during the operation of the process and apply so called data-driven predictive methods. The advantage of using these methods, when compared to FPMs, is the ease of deployment. In contrast to FPM, extensive

knowledge of the modelled process when developing the models is not a must although it can be of advantage if available. Data-driven techniques extract their process knowledge from the measured data automatically by the virtue of their nature and design.

A further development of Soft Sensors may bring numerous additional benefits to the process industry. The main goal of the Soft Sensors is to gain more information about the process. This information may be, for example, a description of the process state which may be extracted from observing a group of relevant measurements. This kind of process state monitoring could provide additional cue for the process operator and may help to predict and, thus, to prevent possible dangerous process states. Another benefit could be the additional information about the product quality. This has to be often evaluated by carrying out expensive laboratory-based analysis which may usually be performed only few times a day. In this case, the Soft Sensor may deliver continuous information stream about the product quality. In a more advanced scenario, Soft Sensors could also be involved in the automated process control loops which would help to increase the plant effectiveness and, thus, for example, reduce the energy consumption of the plant.

In terms of soft sensing, the most commonly applied techniques are Principle Component Regression (PCR) [1] from the statistical methods pool or Artificial Neural Networks (ANN) [2] from the computational intelligence field. Recently, hybrid techniques, which are combinations of the techniques discussed before, have become very popular. Especially neuro-fuzzy methods [3] posses a lot of potential for approaching the solutions of some of the challenges in Soft Sensors modelling. These methods can be easily modified into adapting or evolving methods, which are able to react to changes in the data and thus to change the learnt knowledge base if necessary. There is a large number of evolving neuro-fuzzy methods, for example evolving Takagi-Sugeno (eTS) system [4], [5], Dynamic Evolving Neuro-Fuzzy Inference System (DENFIS) [6] or General Fuzzy Min-Max (GFMM) system [7].

As it was already mentioned, most of the publications dealing with Soft Sensors are based on of either multivariate statistics (e.g. PCA), ANN or neuro-fuzzy approaches to solve process industry related problems. A typical application of Soft Sensors are process monitoring (see for example [8] or [9] for monitoring Soft Sensors based on PCA), prediction of values, which can not be measured on-line (e.g. neural networks based Soft Sensors [10]) or process fault detection Soft Sensors (e.g. [11], [12], [13]). Recently, adaptive Soft Sensors based on evolving neuro-fuzzy methods were published [14].

This work is motived by Jacobs and Jordan [15] [16], where a gating network is used to decide which of the models from a set of available local models, or local experts in the terminology of the cited work, is responsible for the prediction of the given input sample. The predictions of the particular local experts are weighted using weights, which are predicted by the gating networks. In the work of Jacobs and Jordan, there is a special algorithm for the training of the gating networks, which learns and stores the experts responsible for a significant

improvement of the performance of the global model, defined. The algorithm is a kind of winner-takes-all approach which tends to assigns single local experts to partitions of the input space.

2 Gating Artificial Neural Network

In contrast to [15], in this work the responsibility of each of the experts is predicted based on their past performance on similar input samples. The final response of the model is a sum of the expert predictions weighted by the predicted performance of the experts in the current part of the input space. The performance is predicted by the gating Artificial Neural Network (gANN). The aim of the gANN is therefore to learn the performance of the experts in dependency on the position of the input sample in the input space. Thus the input to the gANN are the input samples. The target values of the gANNs has to indicate the performance of the particular experts. The most straight forward way to describe the performance of the experts is to use a measure, which is proportional to the inverted values of the prediction error, for example the Squared Error (SE). The most effective approach to train the gANN is to train one gating network for each of the experts. In this way it is guaranteed that the gANN becomes an *expert* for the performance prediction of the assigned model.

Once trained, the gating networks are able to predict the performance of the particular experts for the test samples x^{test}. Together with the particular predictions, the final response of the model is calculated as:

$$y^p_{final}(x^{test}) = \sum_{i=1}^{N} w_i(x^{test}) y^p_i(x^{test}), \tag{1}$$

where $y^p_{final}(x^{test})$ is the final predicted output of the model given the input test samples x^{test}, y^p_i is the prediction of the ith expert, w_i is the weight of the expert i predicted by the gating network and N the number of available experts.

It is of advantage to apply a feature-selection or PCA/PLS algorithm to the usually high dimensional input data before feeding them to the gating networks. This will limit the input space of the gating networks to the most relevant features and thus allow them to put only the significant patterns of the input space into a relation with the expert's performance.

3 Soft Sensor based on gANN

Based on the approach described in Sect. 2 a Soft Sensor, which is a model combination approach using the gating Artificial Neural Network (gANN), is presented. The structure of the Soft Sensor is shown in Fig. 1. The Soft Sensor consists of a set of experts, which are trained using the labelled training data set $< x^{train}; y^{train} >$. After the training of the experts, next step is the training of the gANN, for this purpose the performance of the experts on a validation data

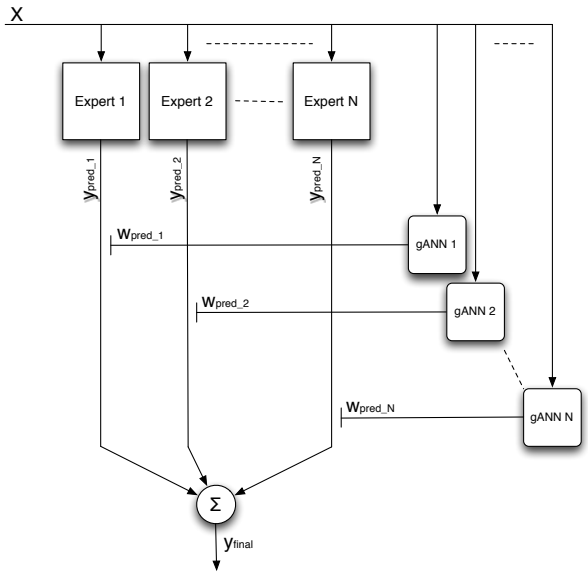


Fig. 1. The structure of the gating Artificial Neural Network based Soft Sensor

set $< x^{val}; y^{val} >$ has to be evaluated. The target values vector for the gANN training w_i^{train} are calculated based on the prediction error e_i^p of the particular experts:

$$w_i^{train} = \frac{1}{1 + \alpha e_i^p} \quad \text{with e.g. } e_i^p = \left(y_i^p - y^{val}\right)^2, \tag{2}$$

where α is a scaling constant which helps to make an efficient use of the range $[0, 1]$, typical values of this constant are in the range $[1, 100]$, y_i^p is the ith expert prediction and e_i^p is the vector of squared prediction errors. The advantage of this performance measure is that, in combination with the scaling constant, it scales the weights to the range $[0, 1]$ automatically. After the calculation of the training weights, the ith gating network can be trained using the labelled data: $< x^{val}, w_i^{train} >$, where x^{val} are the validation data samples.

The experts themselves as well as the gANN are Multi-Layer Perceptrons (MLP) with randomly generated number of hidden units. One has only to specify the range, within which the number of hidden units has to be generated. The advantage of this approach is that one can skip the issue of the a-priori selection of the network topology, because the networks with well-performing topology will automatically get higher weights and be thus prioritised in comparison to experts with less appropriate topology.

Another common issue of ANN and other non-deterministic models solved by this approach is the problem of local minima. Neural Network models are prone to get stuck in local minima during the training and thus achieve a sub-optimal performance on the test data. This is not the case for the proposed

Soft Sensor, because again the sub-optimally performing models will get lower weights assigned.

4 Experiments

This section demonstrates the performance of the proposed Soft Sensor by applying it to the prediction of continuous target values of two industrial data sets.

4.1 Methodology

For the training of the experts and of the gANNs, two-fold cross-validation was used. After running some preliminary experiments, two folds gave the best results. Further on, the term *expert* is used for the set of two networks resulting from the cross-validation, and each of the experts consist of two *partial-experts*. The partial-experts are trained using labelled training data set $Z^{train} := < x^{train}, y^{train} >$. After the training the performance of the partial-experts is evaluated using the exclusive validation data Z^{val}. The validation results of the partial-experts represent the target values of the training data for the gating networks $Z^{gateTrain}$. For the evaluation of the gANN, there is another exclusive validation data partition $Z^{gateVal}$ necessary. The gANN validation data is the same for both of the gating networks from the cross-validation, which guaranties that both gating networks are assessed using the same independent data. The last partition of the data is the test data Z^{test}, which is being used for the performance evaluation of the whole Soft Sensor. The partitioning of the data is presented graphically in Fig. 2.

The parameters of the data partitioning for the experiments are the following, for the cross-validation (Z^{train}+Z^{val}) 50% of the data samples has been used, which means that 25% of the samples are used for the actuall training Z^{train} and the other 25% of the total number of samples for the validation Z^{val} of the particular CV-folds. The gate validation set $Z^{gateVal}$ are another 20% of the data and the remaining 30% were alocated for the test purposes. Because of the

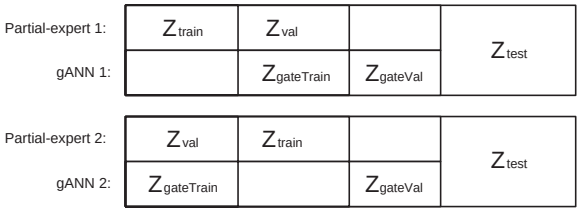


Fig. 2. Partitioning of the data to the training, gate validation and test data.

application of the cross-validation, there is two degrees of freedom for combining

the results of the partial-experts. Firstly, one can combine the partial-experts to obtain the experts in different ways. The traditional approach is averaging the partial-experts predictions. Additionally, the presented approach allows to build a weighted sum of the partial-experts by using the weights predicted by the gANN. The second degree of freedom for building the combinations is at the level of the experts. The aim of the proposed approach is to build a set of experts and combine them to a final prediction. For the experiments the following combination types were considered:

Table 1. Considered model combinations approaches

	partial-experts combination	experts combination
Type 1	Mean	Mean
Type 2	Random selection	Weighted
Type 3	Weighted	Best performance
Type 4	Weighted	Weighted

Type 1: This is the traditional approach, where to obtain the prediction of the cross-validation ensemble, the mean value of the individual partial-expert predictions is built. For the experts combination the same is done, namely an average over the responses of the experts is built. As it is the simplest way of combining the models without involving the weights from the gating networks, this method represents the performance base-line for the comparison with the other methods.

Type 2: In this case, the combination at the level of the partial-experts is done by randomly selecting one of both partial-experts. The experts combination is a weighted sum of the experts, where the weights are obtained from the gating networks.

Type 3: Here, the experts are built as weighted sums of the partial-expert predictions. At the expert level, the selected expert is the one with the best performance on the gate validation data set $Z^{gateVal}$, which corresponds to the winner-takes-all approach.

Type 4: This is the approach discussed in Section 2 and Section 3. At the cross-validation level as well as at the expert level the output is a weighted sum of the individual predictions.

4.2 Results of the drier Soft Sensor[1]

The drier Soft Sensor was developed using the methodology described in Section 4.1. Because of the high dimensionality of the input data it turned out to be of

[1] Data set provided by Evonik Degussa AG

advantage to limit the input space of the gating networks. This has been achieved by applying the PCA algorithm [1] and taking the first five PCA features for further use.

There were 100 experts simulated. The number of hidden units of the experts was generated randomly within the range $[1, 10]$ by sampling from an uniform distribution. For each partial-experts a set of five different gANN with random number of hidden units (within the range $[6, 14]$) was trained. The performance of the gANN was assessed using the $Z^{gateVal}$ data partition. The gating network with the best performance on that data was selected and stored for modelling the weights.

The following results evaluation focus on the MSE performance as a function of increasing number of involved experts. Fig. 3 compares Mean Squared Errors of the Type1 to Type4 models, as a function of the number of involved experts. One can see that, with an exception the winner-takes-all method, each

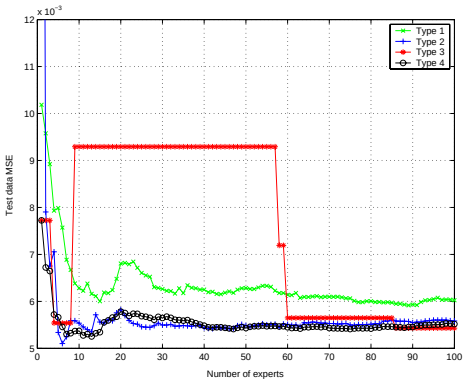
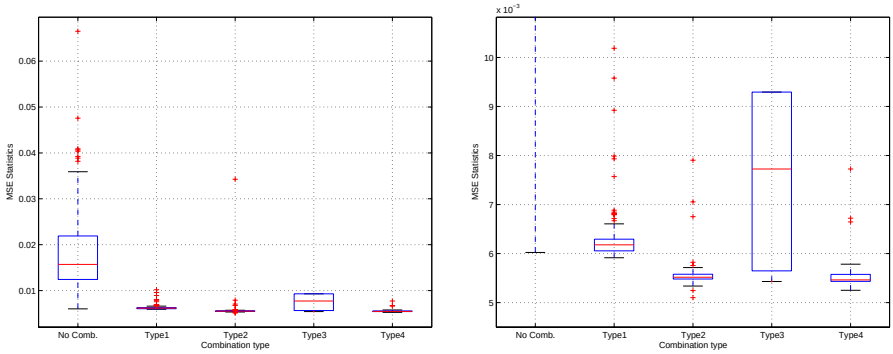


Fig. 3. The MSE performance of the drier Soft Sensor as a function of the number of experts N

of the methods converges with increasing number of involved experts to a stable performance level. The convergence value of the approaches using the weights of the gANN (Type2,3,4) are in general lower than the convergence of the base-line approach. In case of the winner-takes-all approach, there cannot be any convergence guaranteed, because in this case the output corresponds to the expert with the best performance on the validation data set, but in general this does not necessarily correspond to the best performance on the test data set. Fig. 4 presents the MSE statistics of the particular combination methods and the individual partial-experts in form of boxplots. From Fig. 4a it is obvious that there is large variance in the performance of the partial-experts. This shows one of the advantages of model combination approaches, namely the *stabilisation* of the results. In Fig. 4b one can see that besides of few outliers the performance of the weighted combinations (Type2 and Type4) is better than the performance of

the best individual partial-expert. Additionally, Fig. 4b shows that in terms of convergence speed (median value of the boxplots) and stability (size of the boxplots) the Type2 and Type4 combination methods achieve superior performance compared to the base-line combination approach (Type1).



(a) Full view (b) Zoomed view

Fig. 4. Statistics of the combination approaches together with the performance of the single partial-experts

4.3 Results of the debutanizer column Soft Sensor[2]

Again, the methodology described in Section 4.1 was applied to develop this Soft Sensor. For the gating networks partial correlation based feature selection (see e.g. [10]) was applied. There were 200 experts built and combined for the Soft Sensor. The number of hidden units of the experts was generated randomly within the range $[1, 10]$. For each partial-expert a set of five different gANN with random number of hidden units (within the range $[6, 14]$) was trained, from this set the best gANN in terms of the gate-validation data was selected for further processing.

The dynamics of the MSE shows again convergence of the performance towards a stable level, as can be observed in Fig. 5. If assuming the distributions of the particular curves from Fig. 5 as normal and having the same standard deviation, then compared to the base-line approach (Type1) the approaches Type2 and Type4 achieve a significant performance gain.

From the boxplot statistics presented in Fig. 6, one can again observe high variance of the performances of the individual partial-experts. Also in this case study, the approaches involving the gating networks outperform both the individual partial-experts and the base-line combination method.

[2] Data set available at: www.springer.com/1-84628-479-1

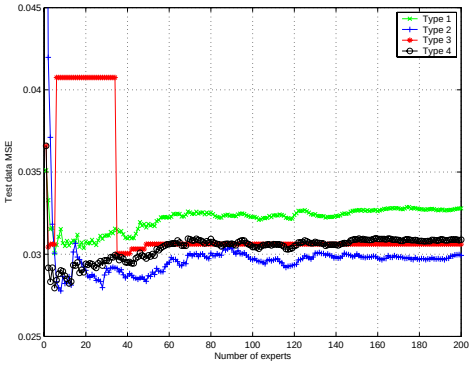
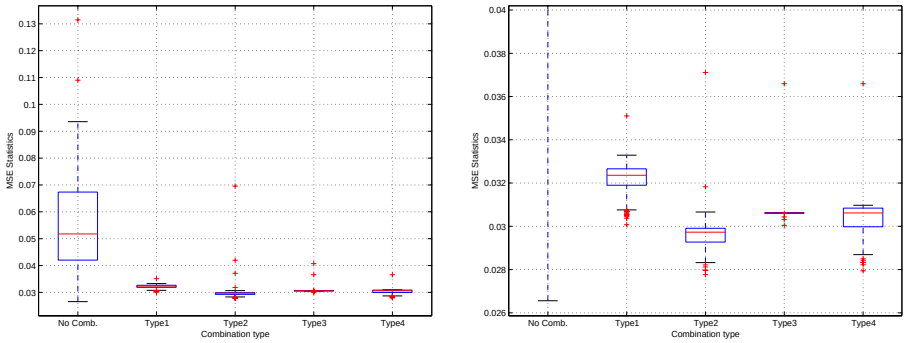


Fig. 5. The MSE performance of the debutan Soft Sensor as a function of the number of experts N



(a) Full view (b) Zoomed view

Fig. 6. Statistics of the combination approaches together with the performance of the single partial-experts

5 Summary

Training a set of models, or experts, and combing their predictions has in context of process industry data proven as a powerful approach to handle two issues of the traditional modelling practice, namely the a-priori selection of best model parameters and the handling of local minima problem. The a-priori model parameter, like the number of hidden units in the case of an ANN model, selection is handled by using so called gating networks, which are trained to predict the performance of the experts. These networks will predict lower weights for experts with lower performance and thus decrease their influence on the final prediction. The problem of local minima is solved in the same way. When a model gets stuck in a local minimum during the training, it will achieve sub-optimal performance

on the validation and test data and the gating network will automatically assign lower weights to such an expert.

Although there are single models in the expert pool, which achieved better performance, because of the problems discussed before one cannot rely on the fact, that these optimally performing models will be identified during the training phase. The proposed approach performed better than both, the average expert in the pool and the base-line approach to model combination, namely the mean building of the prediction of the experts.

References

1. Jolliffe, I.T.: Principal Component Analysis. Springer (2002)
2. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, USA (1995)
3. Jang, J.S.R., Sun, C.T., Mizutani, E.: Neuro-fuzzy and soft computing. Prentice Hall Upper Saddle River, NJ (1997)
4. Angelov, P.P., Filev, D.P.: Flexible models with evolving structure. International Journal of Intelligent Systems **19**(4) (2004) 327–340
5. Angelov, P.P., Filev, D.P.: An approach to online identification of takagi-sugeno fuzzy models. Systems, Man and Cybernetics, Part B, IEEE Transactions on **34**(1) (2004) 484–498
6. Kasabov, N.K., Song, Q.: Denfis: dynamic evolving neural-fuzzy inference system and itsapplication for time-series prediction. Fuzzy Systems, IEEE Transactions on **10**(2) (2002) 144–154
7. Gabrys, B., Bargiela, A.: Neural networks based decision support in presence of uncertainties. Journal of Water Resources Planning and Management **125**(5) (1999) 272–280
8. Champagne, M., Dudzic, M., Inc, T., Temiscaming, Q.: Industrial use of multivariate statistical analysis for process monitoring and control. American Control Conference, 2002. Proceedings of the 2002 **1** (2002)
9. Li, W., Yue, H.H., Valle-Cervantes, S., Qin, S.J.: Recursive pca for adaptive process monitoring. Journal of Process Control **10**(5) (2000) 471–486
10. Fortuna, L.: Soft Sensors for Monitoring and Control of Industrial Processes. Springer (2007)
11. Dunia, R., Qin, J., Edgar, T.F., McAvoy, T.J.: Sensor fault identification and reconstruction using principal component analysis. In: Proceedings of the 13 thTriennial World Congress. (1996) 259–264
12. Dunia, R., Qin, S.J.: Joint diagnosis of process and sensor faults using principal component analysis. Control Engineering Practice **6**(4) (1998) 457–469
13. Amazouz, M., Pantea, R.: Use of multivariate data analysis for lumber drying process monitoring and fault detection. In Crone, S.F., S., L., Stahlbock, R., eds.: International Conference on Data Mining. (2006) 329–332
14. Macias, J.J., Zhou, P.X.: A method for predicting quality of the crude oil distillation. In: Evolving Fuzzy Systems, 2006 International Symposium on. (2006) 214–220
15. Jordan, M.I., Barto, A.G.: Task decomposition through competition in a modular connectionist architecture: The what and where vision tasks. COGNITIVE SCIENCE **15** (1991) 219–250
16. Jacobs, R.: Adaptive mixtures of local experts. Neural Computation **3**(1) (1991) 79–87